

Cell Volume Control at a Surface for Three-Dimensional Grid Generation Packages

Stephen J. Alter^{*}

Lockheed Engineering & Sciences Company, Hampton, Virginia

and

Kenneth J. Weilmuenster[†]

NASA Langley Research Center, Hampton, Virginia

Abstract

This paper presents an alternate method of calculating the cell size for orthogonality control in the solution of Poisson's three-dimensional space equations. The method provides the capability to enforce a better initial guess for the grid distribution required for boundary layer resolution. This grid point distribution is accomplished by enforcing grid spacing from a grid block boundary where orthogonality is required. The actual grid spacing or cell size for that boundary is determined by the two or four adjacent boundaries in the grid block definition, which are two dimensional grids. These two dimensional grids are in turn defined by the user using insight into the flow field and boundary layer characteristics. The adjoining boundaries are extended using a multi-functional blending scheme, with user control of the blending and interpolating functions to be used. This grid generation procedure results in an enhanced Computational Fluid Dynamics calculation by allowing a quicker resolution of the configuration's boundary layer and flow field and by limiting the number of grid re-adaptions. The cell size specification calculation has been applied to a variety of configurations ranging from axisymmetric to complex three-dimensional configurations. Representative grids will be shown for the Space Shuttle and the Langley Lifting Body (HL-20).

Introduction

Design and development of aerodynamic configurations in industry is primarily being done with inviscid solvers as well as intricate panel methods. [1, 2, 3] These simplified solvers are being employed as methods of homing in on optimum designs for a variety of contracts

^{*}Aeronautical and Astronautical Engineer

[†]Senior Research Engineer, Aerothermodynamics Branch, Space Systems Division, Member AIAA

and developmental programs. Yet most contractors find it necessary to further optimize the designs through detailed studies and gathering of experimental data [1].

Although the design processes employed by industry optimize configurations based on designed mission profiles, the capabilities for acquiring experimental data at these design points are limited. As a result, Computational Fluid Dynamics (CFD) is being used as an alternative source of detailed evaluation data. Most users in the CFD community would prefer to use CFD codes modeling viscous effects in the design process because they model the non-linear effects of the flow characteristics [4]. By utilizing this type of CFD code, a more competitive environment is established for design optimization and resulting in better configuration designs. However, current CFD codes can not be used as design tools, simply because they are still too time consuming to be used in parametric studies.

Current research and development in the CFD arena has been focusing on accurate simulation of viscous flows [5, 6, 7]. The area of most development has been in the generation of Navier-Stokes solvers, with emphasis on the algorithms employed as opposed to the resolution of the grid for numerical accuracy. As progress has been made in the generation of these viscous flow solvers, more complicated configurations are being envisioned and designed, especially in the area of high speed aerodynamics. Some typical complex configurations include the National Aerospace Plane [8] to the advanced fighter designs [9]. The increased complexity of configurations has led to more complex solution algorithms in order to gain numerical accuracy.

There are several different methods for obtaining numerical accuracy based on the grid. One way is to re-adapt the grid being used based on flow characteristics. This type of procedure will allow for most efficient use of the available grid points. With the emergence of the SAGE code [10] to the shock alignment procedure that is used in the Langley Aerothermodynamic Upwind Relaxation Algorithm (LAURA) code [11], numerical accuracy via grid re-adaption is being made possible. By obtaining numerical accuracy in this fashion, the number of grid points necessary to solve the flow field calculations can be reduced with a corresponding rapid resolution of the flow characteristics. Furthermore, these re-adaption codes assume a solution has been generated and grid adaptation is done to improve numerical accuracy and convergence.

However, the original grids used by the solvers to obtain an initial solution for grid adaptation have grids that are coarse near the surface, or wall. The coarse spacing typically results from the methods used to determine the cell sizes necessary to calculate the forcing functions of the elliptic solvers. The most commonly used method for determining cell sizes for orthogonality control is Trans Finite Interpolation (TFI). TFI is based on maintaining the overall grid distribution as defined by the boundaries of the grid block flow domain or surface edges. Yet, as a result of the dependency on the opposing flow domain boundaries, the grid lines that result may have a highly skewed appearance (figure 1). This skewness of grid lines from the surface will cause the overall cell size to be much larger as opposed to the grid line being orthogonal to the surface. This leads to excessive grid re-adaptations to resolve the boundary layer.

In order to generate a quality grid for a Navier-Stokes CFD calculation, the grid distribution must allow for the accurate resolution of the viscous forces at a solid boundary. The grid spacing is controlled in elliptic solvers by the specification of the first cell sizes off of the configuration's surface and the decay rate of the forcing functions for orthogonality; this pa-

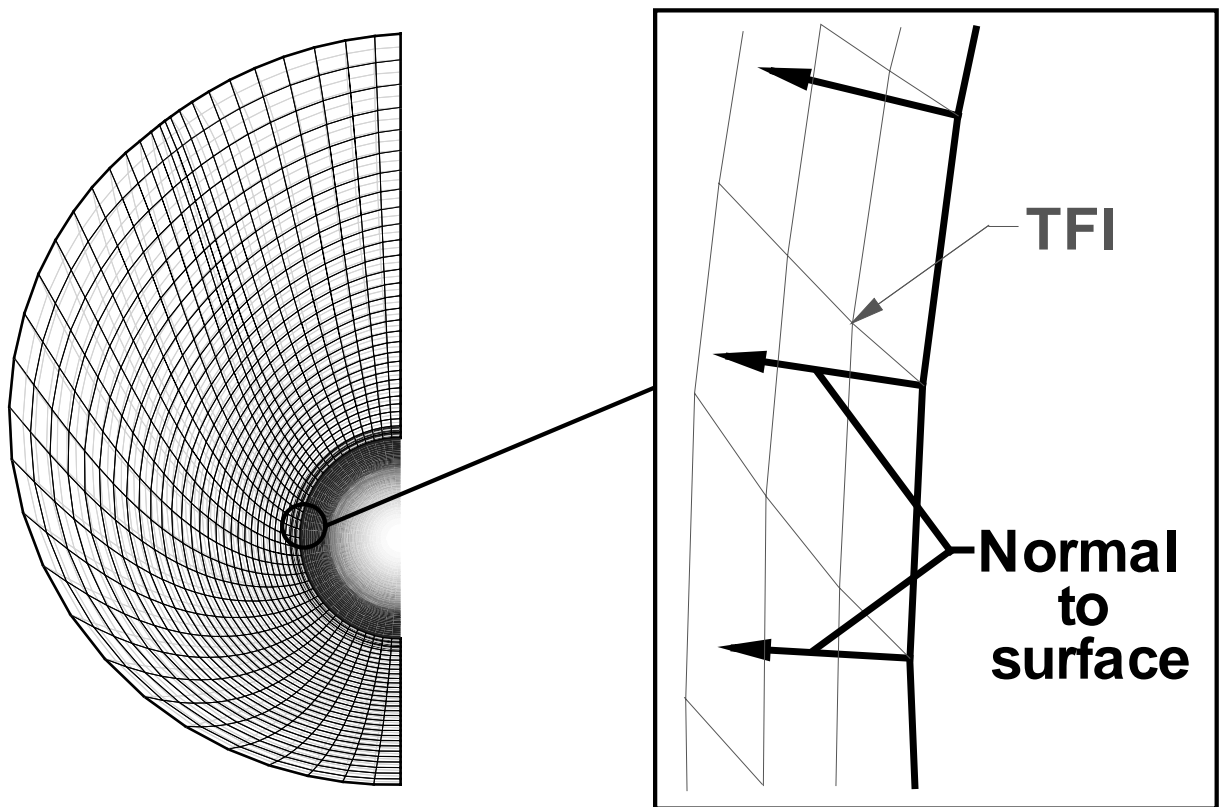


Figure 1: Grid line skewness produced by TFI.

per will only consider the latter. Thus a better guess as to the surface cell sizes necessary to model the boundary layer flow characteristics, with the use of local grid densities is needed.

The areas that require high grid densities and small cell sizes, can be identified using several methods. One might be able to use current databases of experimental data for configurations having similar geometric characteristics and evaluated under similar flight conditions. Another might be to use linear panel theory. Nonetheless, the general areas of concern for proper boundary layer modeling can be determined and by having proper surface and near-surface grid densities, accuracy and solution convergence rates can be maximized.

The purpose of this paper is to present an alternate method that links the problem of a better initial grid to the solving of Poisson's three dimensional space equations. The method allows the user to use insight to determine the "best" guess of the local cell size for different computational regions and define the cell sizes on the surface domain with a multi-functional blending scheme. This blending is done by smearing the Local ARc length Cell Sizes (LARCS) based on the configuration's surface edges, into two distinct planes that represent the first interior points from the configuration's surface. Then a continuous hyperbolic function is used to combine these interior planes to generate a single plane representing the cell sizes off the configuration's surface. The blending will result in a smooth cell size transition between all defining surface edges, as well as removing the dependency on the opposing grid block boundary.

Method

The overall goal of this method is to take the best guess of the cell size to be used in given areas of a configuration for boundary layer resolution, and propagate the cell sizes onto the configuration surface. The resulting blend will be smooth from computational region to computational region. This process begins by the determining the local cell sizes of the grid to use on the defining boundaries of the grid blocking structure (figure 2).

The local wall cell size Δs_w , given in equation 1

$$\Delta s_w = \frac{Re_w \mu_w}{\rho_w a_w} \quad (1)$$

is based on the cell Reynold's Number at the surface. By setting the local cell Reynold's number to a value of 2.0 or less [11], the proper cell size can be determined (figure 3). This Reynold's number and the resulting cell size, will enable a CFD code to capture the viscous forces in the boundary layer [15, 16]. By accomplishing this capture early in the CFD calculation of the viscous flow field, enhanced convergence of the boundary layer will occur resulting in a reduced number of successive grid re-adaptions. Hence, the flow resolution should be expedited.

The other parameters required to calculate the cell sizes include the local temperature (T_w) and density (ρ_w). These parameters can be determined by any number of methods. Some of these include comparing experimental or computational results of a similar configuration, using some of the relationships for ideal or perfect gases [12] or a simple impact theory program such as APAS [3]. In any case the overall goal is to determine a local density, and temperature for a given region on the surface of the configuration.

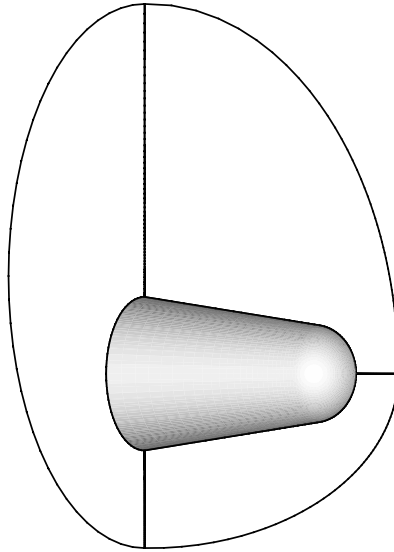


Figure 2: Block structure to be utilized in defining flow field domain.

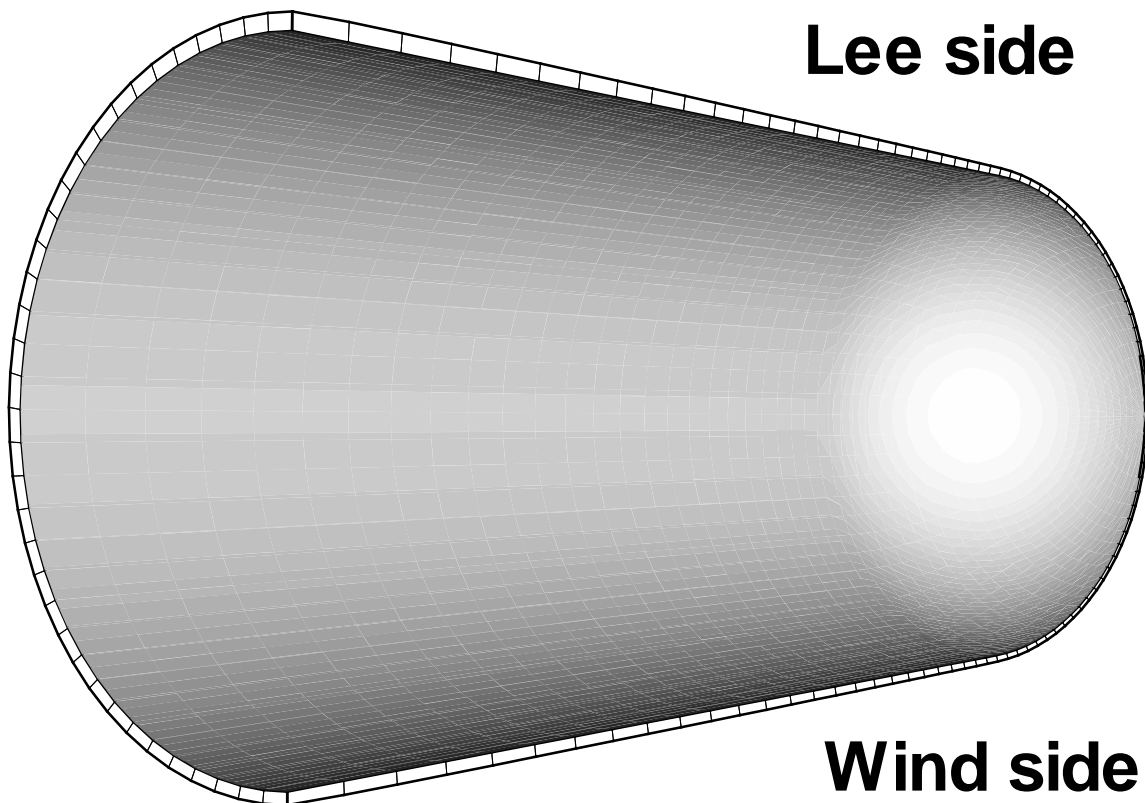


Figure 3: Cell size on a surface based on an initial guess with large cell sizes on the leeside decreasing towards the windside.

The temperature is used to calculate two other needed parameters. The first is the local speed of sound (equation 2)

$$a_w = \sqrt{\gamma RT_w} \quad (2)$$

used in the cell Reynold's number equation. The second parameter that is derived from temperature is the viscosity. It can be determined by using Sutherland's Law [13], equation 3 or reference [14] for those cases where temperatures are above 2000deg K.

$$\mu_s = 1.4584 \times 10^{-5} \frac{(T^\circ K)^{3/2}}{(T^\circ K) + 110.33} \quad (3)$$

Thus the initial cell sizing parameter calculation is dependent on the user's discretion, but the resulting cell size should represent the type of flow characteristics expected. For example, if a cone was being evaluated at an angle of attack of 40 degrees, one would expect cell sizes on the leeward side to be large compared to the windward side. (figure 3.)

After determining the cell sizes to be utilized in the different regions of the block boundaries, a grid generator such as GRIDGEN [17], PATRAN[18], etc., is used to define the two-dimensional grids for each face of the grid block (figure 4). Then three-dimensional Poisson solvers [19] are employed to create the volume grid for computational analysis. In order to solve these equations to obtain orthogonality on the block boundary surfaces of choice, the cell size, computed by equation 4,

$$\Delta s = \sqrt{\Delta x^2 + \Delta y^2 + \Delta z^2} \quad (4)$$

has to be known. It is this cell size that LARCS is used to determine.

The determination of the cell sizes emanating from any given point on a configuration's surface by LARCS begins with the functionalization of the cell sizes of the adjacent boundaries with respect to their parametric coordinates. For example, if cell sizes for the surface of the configuration in figure 5 are being determined, the defining boundaries are the leeside and the windward side symmetry planes (faces 3 and 4 respectively), the pole boundary (face 1) and the exhaust plane (face 2). Next the cell size as a function of parametric coordinate is determined for each edge (figure 6.) using the standard arc-length equation 4.

From the functionalization, two distinct planes representing cell sizes off of the configuration's surface are created. The first plane is generated by blending from the cell size functionalization on face 1 to that of face 2, in the ξ direction. Similarly, the second plane is a blend between face 3 and face 4 in the η direction. These planes can be generated utilizing linear or parabolic blending. For the linear blending of faces 1 and 2 (figure 5), the basic blending coefficient calculated via equation 5

$$\sigma_\xi = \frac{\xi_{max} - \xi}{\xi_{max} - 1} \quad (5)$$

with the condition in equation 6

$$\sigma_{\xi_{1-2}} + \sigma_{\xi_{2-1}} = 1.0 \quad (6)$$

forms the blending coefficient of face 1 to face 2, equation 7

$$\sigma_{\xi_{1-2}} = \frac{\xi_{max} - \xi}{\xi_{max} - 1} \quad (7)$$

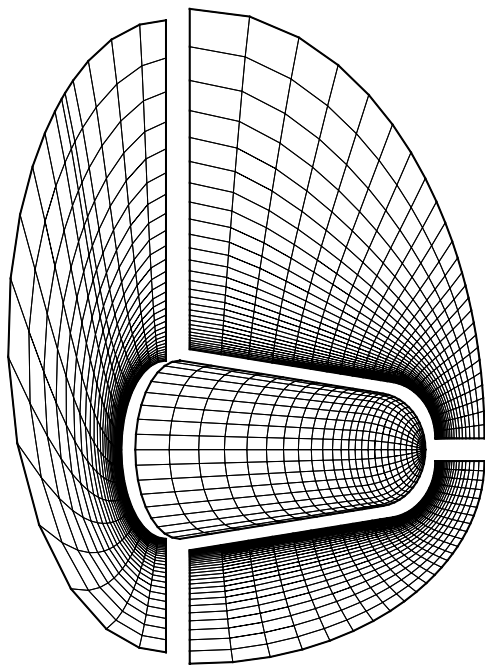


Figure 4: Two dimensional grid definitions on flow domain block boundaries.

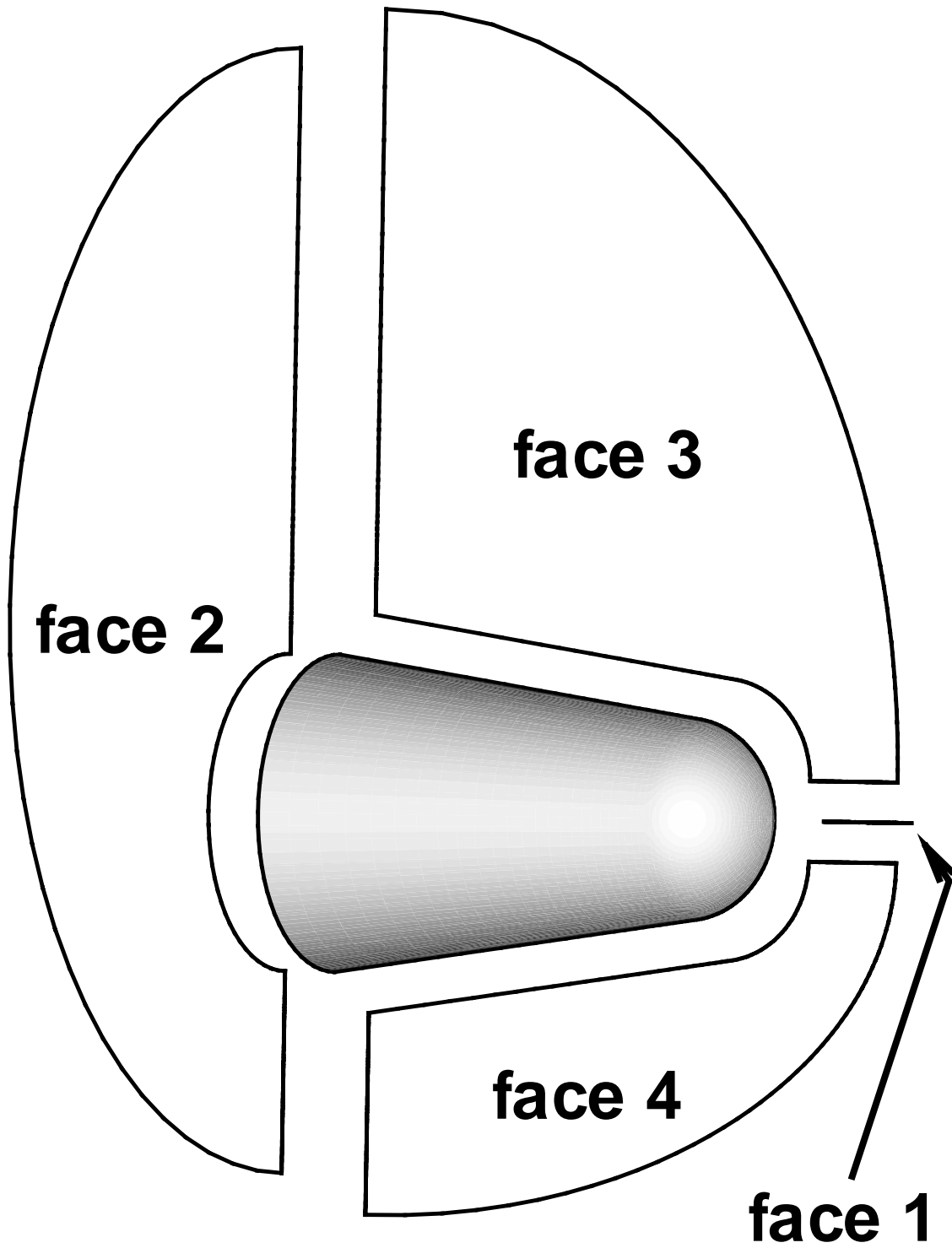
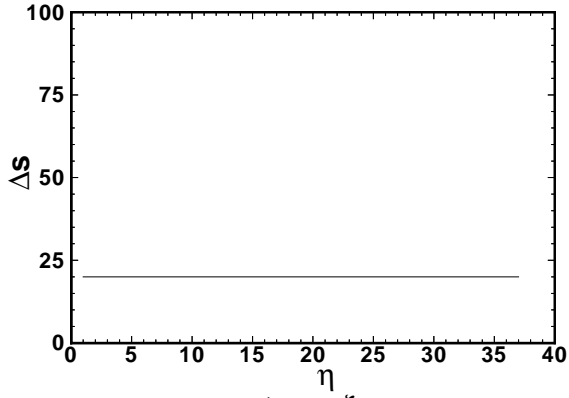
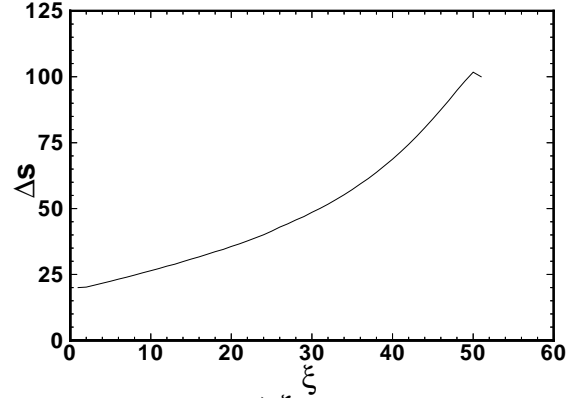


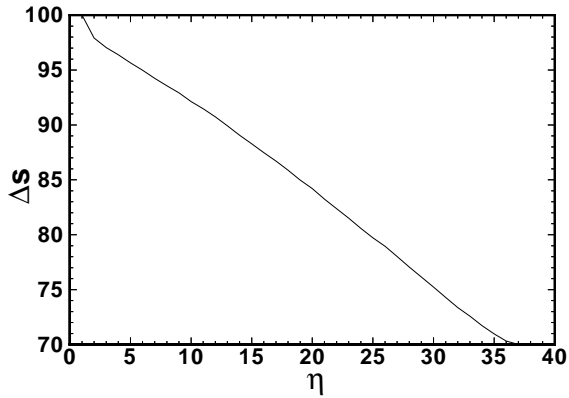
Figure 5: Simple cone for cell sizing calculation.



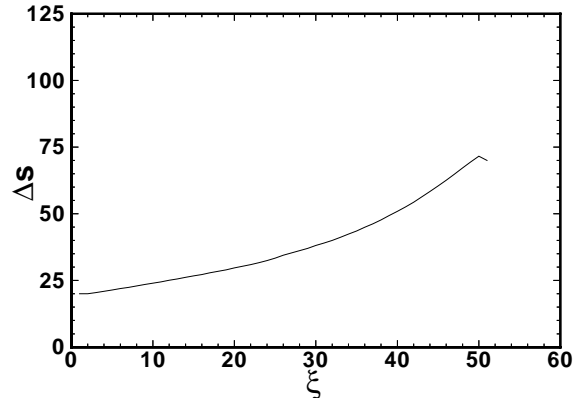
a) η at ξ_{\min}



c) ξ at η_{\min}



b) η at ξ_{\max}



d) ξ at η_{\max}

Figure 6: Cell sizing functionalization based on parametric coordinate for each edge of the configuration's surface.

and the blending coefficient of face 2 to 1, equation 8.

$$\sigma_{\xi_{2 \rightarrow 1}} = 1 - \frac{\xi_{max} - \xi}{\xi_{max} - 1} \quad (8)$$

Using the linear blending, equation 9

$$\Delta s_{\xi, \eta} = \sigma_{\xi_{1 \rightarrow 2}} \Delta s_{\xi_{min}} + \sigma_{\xi_{2 \rightarrow 1}} \Delta s_{\xi_{max}} \quad (9)$$

substituting in equations 7 & 8, and combining like terms, equation 10 is used to blend from opposing faces 1 and 2.

$$\Delta s_{\xi, \eta_{1 \rightarrow 2}} = \left(\frac{\xi_{max} - \xi}{\xi_{max} - 1} \right) (\Delta s_{\xi_{min}} - \Delta s_{\xi_{max}}) + \Delta s_{\xi_{max}} \quad (10)$$

Likewise, linear blending of opposing faces 3 and 4 can be written as in equation 11 (figure 7).

$$\Delta s_{\xi, \eta_{3 \rightarrow 4}} = \left(\frac{\eta_{max} - \eta}{\eta_{max} - 1} \right) (\Delta s_{\eta_{min}} - \Delta s_{\eta_{max}}) + \Delta s_{\eta_{max}} \quad (11)$$

The parabolic blending is somewhat more complicated but can be derived the same way with the basic blending coefficient as described in equation 5 and the condition stated in equation 6. However, the blending coefficients $\sigma_{\xi_{1 \rightarrow 2}}$ and $\sigma_{\xi_{2 \rightarrow 1}}$, are different due to their parabolic nature; $\sigma_{\xi_{1 \rightarrow 2}}$ becomes:

$$\sigma_{\xi_{1 \rightarrow 2}} = \frac{\sigma_{\xi}^2}{1 - 2\sigma_{\xi} + 2\sigma_{\xi}^2} \quad (12)$$

and $\sigma_{\xi_{2 \rightarrow 1}}$ becomes:

$$\sigma_{\xi_{2 \rightarrow 1}} = \frac{1 - 2\sigma_{\xi} + \sigma_{\xi}^2}{1 - 2\sigma_{\xi} + 2\sigma_{\xi}^2} \quad (13)$$

Thus, the parabolic blending for face 1 to 2, is given by equation 14,

$$\Delta s_{\xi, \eta_{1 \rightarrow 2}} = \frac{\sigma_{\xi}^2 \Delta s_{\xi_{min}} + (1 - \sigma_{\xi})^2 \Delta s_{\xi_{max}}}{1 - 2\sigma_{\xi} + 2\sigma_{\xi}^2} \quad (14)$$

and the parabolic blending from faces 3 to 4 (figure 8) is given by equation 15.

$$\Delta s_{\xi, \eta_{3 \rightarrow 4}} = \frac{\sigma_{\eta}^2 \Delta s_{\eta_{min}} + (1 - \sigma_{\eta})^2 \Delta s_{\eta_{max}}}{1 - 2\sigma_{\eta} + 2\sigma_{\eta}^2} \quad (15)$$

In either case, the resulting surfaces from the linear blending, figures 9 and 10, as well as the surfaces from the parabolic blending, figures 11 and 12 represent two unique surfaces for specifying cell sizes for orthogonality control. For most symmetry planes this may indeed be true because the user may only want blending between the configuration surface and the outer flow domain. Yet for the block surfaces that depend more heavily on all four edges, these two artificial surfaces have to be combined. This will enable the attributes of each edge to affect the overall cell sizes used in the specification of orthogonality.

To generate a surface which has attributes of each of the four functionalized edges, the two artificial surfaces are combined with a hyperbolic blending scheme. This scheme becomes evident when viewing the configuration surface domain in parametric space. Consider figure

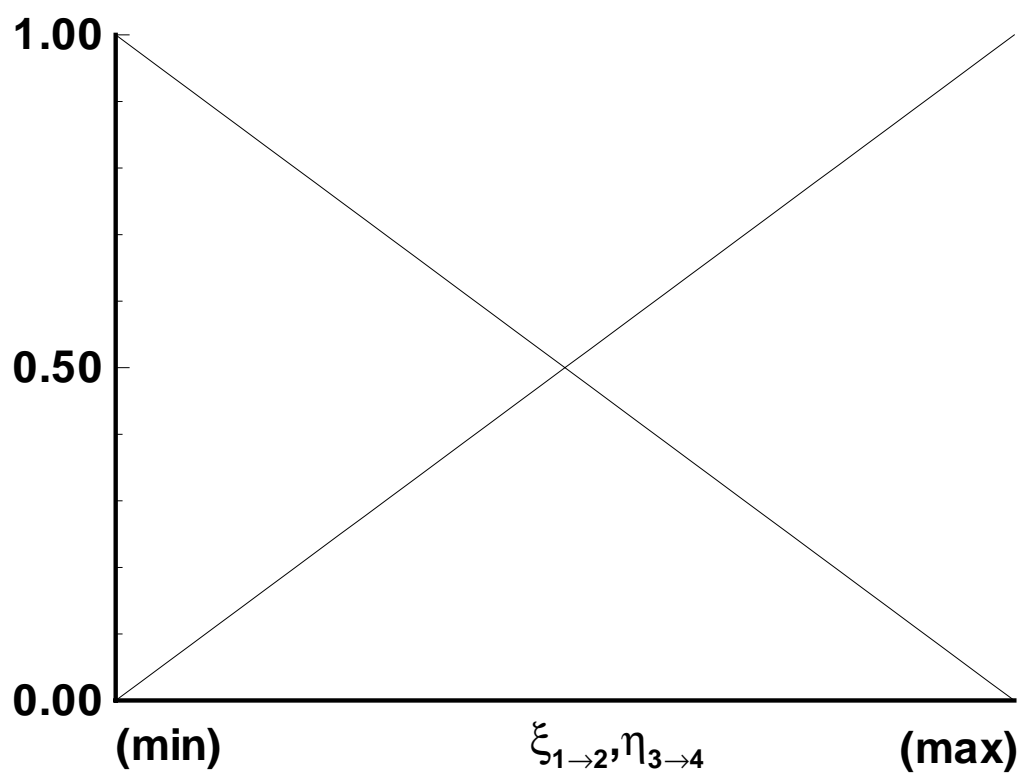


Figure 7: Linear blending functions between opposing faces 1 & 2 and 3 & 4.

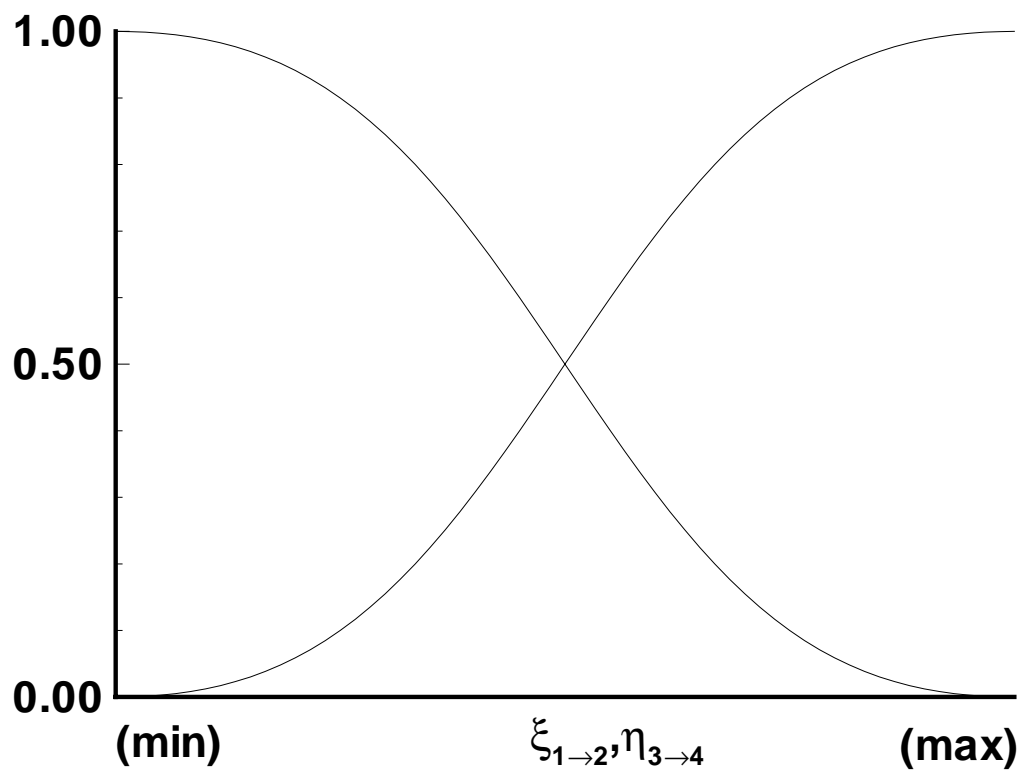


Figure 8: Parabolic blending functions between opposing faces 1 & 2 and 3 & 4.

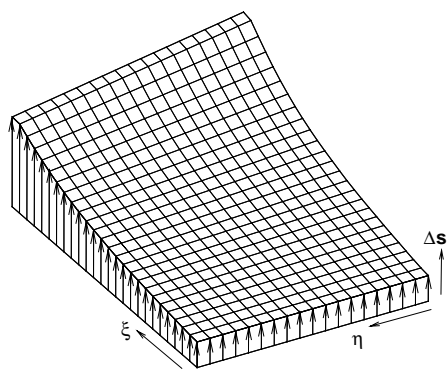


Figure 9: Artificial surface results from linearly blending face 1 to 2.

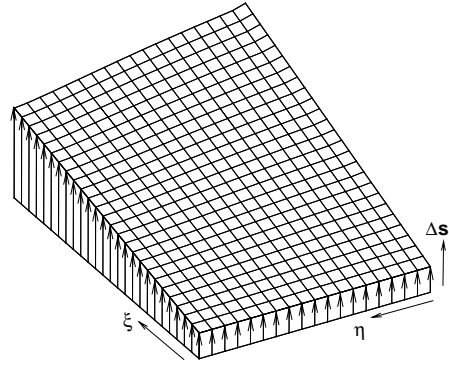


Figure 10: Artificial surface results from linearly blending face 3 to 4.

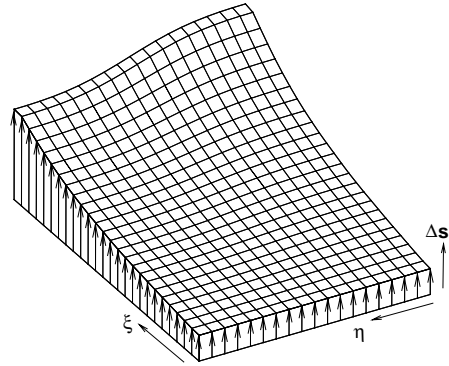


Figure 11: Artificial surface results from parabolically blending face 1 to 2.

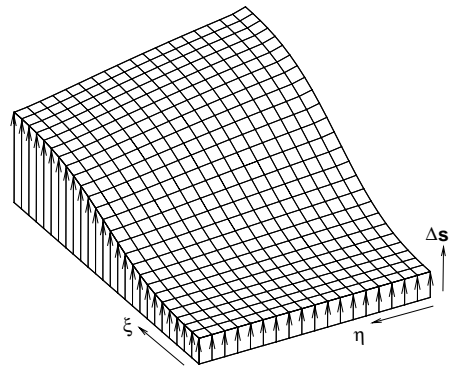


Figure 12: Artificial surface results from parabolically blending face 3 to 4.

13, where the percentages of the four edge functions are to be used to calculate the overall cell sizes. For the corners of the domain, the resulting cell sizing surface is made up of 50% of each function that meets at the respective corner, or the two individual artificial surfaces.

In addition, the center of the domain has 50% of each artificial surface. Also notice that at the middle of each edge, there is either a 100% or a 0% of the surface to be utilized. This arises merely from the initial linear and/or parabolic blending of the opposing face edge functions. For the surface that contains the actual edge function, the dominant edge, 100% of that function is to be use on that edge. Likewise, the other surface, the non-dominant edge function is to have no affect on the resulting hyperbolically blended surface, because it is either a line or a result of the initial parabolic blend.

By connecting the 50% regions with straight lines in the parametric domain the hyperbolic structure can be seen. Equations 16 and 17 represent the blending functions for face 1 to 2 and face 3 to 4 respectively, as shown in figure 14.

$$\delta_\xi = \frac{1}{2}|\alpha^2 - \beta^2 - 1| \quad (16)$$

$$\delta_\eta = \frac{1}{2}|\beta^2 - \alpha^2 - 1| \quad (17)$$

Here,

$$\alpha = 2 \left(\frac{\xi_{max} - \xi}{\xi_{max} - 1} \right) - 1$$

and,

$$\beta = 2 \left(\frac{\eta_{max} - \eta}{\eta_{max} - 1} \right) - 1 \quad .$$

Combining these blending coefficients, and hence the initial artificial surfaces, equation 18 results.

$$\Delta s_{\xi,\eta} = \delta_\xi \Delta s_{\xi,\eta_{1 \rightarrow 2}} + \delta_\eta \Delta s_{\xi,\eta_{3 \rightarrow 4}} \quad (18)$$

In addition, the plot of the percentages in parametric space represent two distinct blending function surfaces, one for each opposing face pair. Thus for any parametric point on the surface, certain percentages of each surface is used, and the combined percentages sum to 100%. Due to the smoothness of the blending function, the effects of all four edges are smeared throughout the resulting blended surface's domain.

By utilizing this function, the overall cell size necessary for orthogonality specification can be calculated and is based on the defining edges of the configuration's surface domain. The resulting blended surface as shown in figure 15 is a better description for cell size because it now limits the maximum distance a grid point can be from the configuration surface and is not as dependent on surface curvature as other methods used. In effect, a multi-functional blending scheme is used to determine the cell sizes. Thus, by utilizing the cell sizes determined by the LARCS method, the forcing functions necessary to control orthogonality within elliptic volume grid generators are more dependent on the users best guess. In doing so, the control of grid spacing near the wall is more related to the problem at hand, as opposed to the rapid solution of the elliptic Poisson's equations.

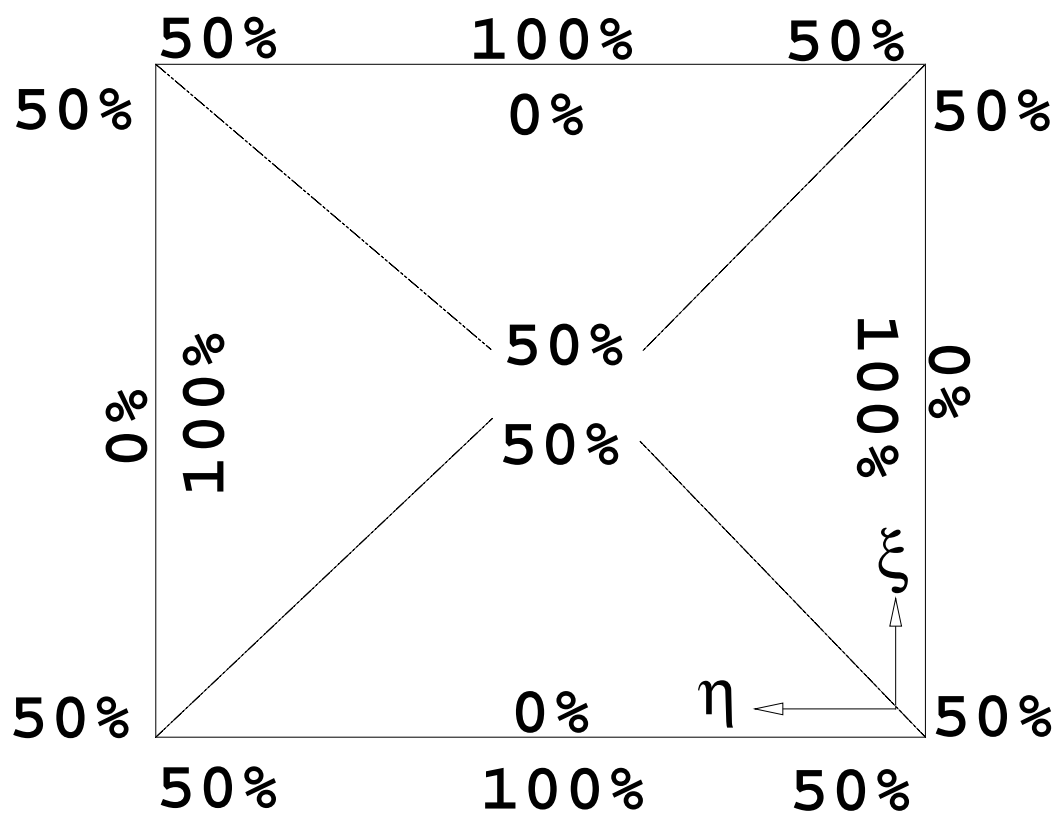


Figure 13: Percentages of surface functions in the parametric domain.

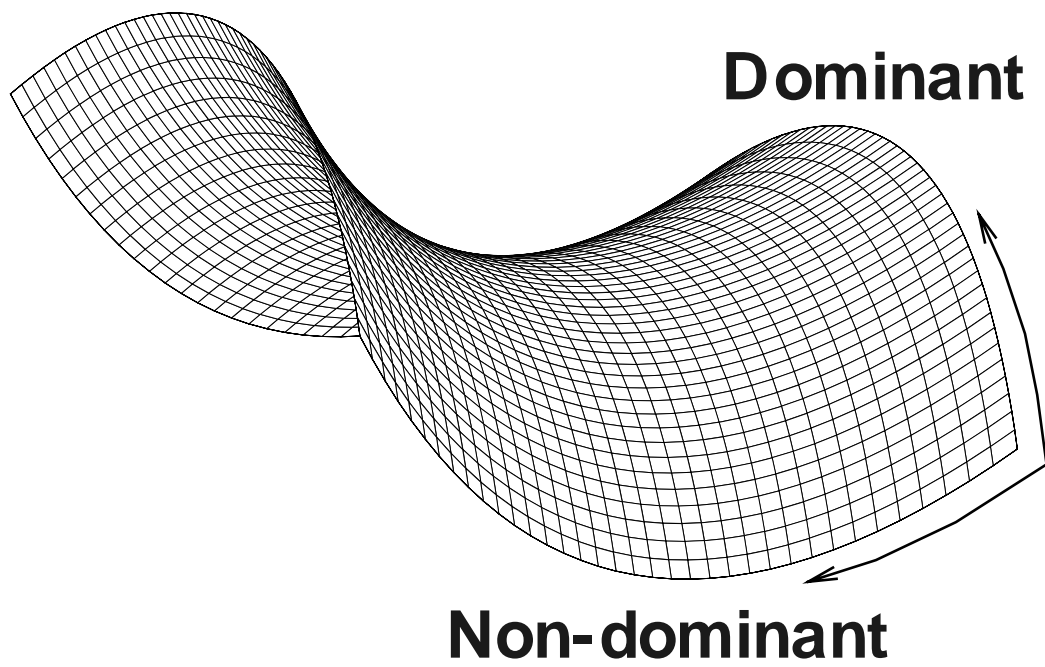


Figure 14: Hyperbolic blending function in parametric space for the dominant and non-dominant edge cell sizing functions.

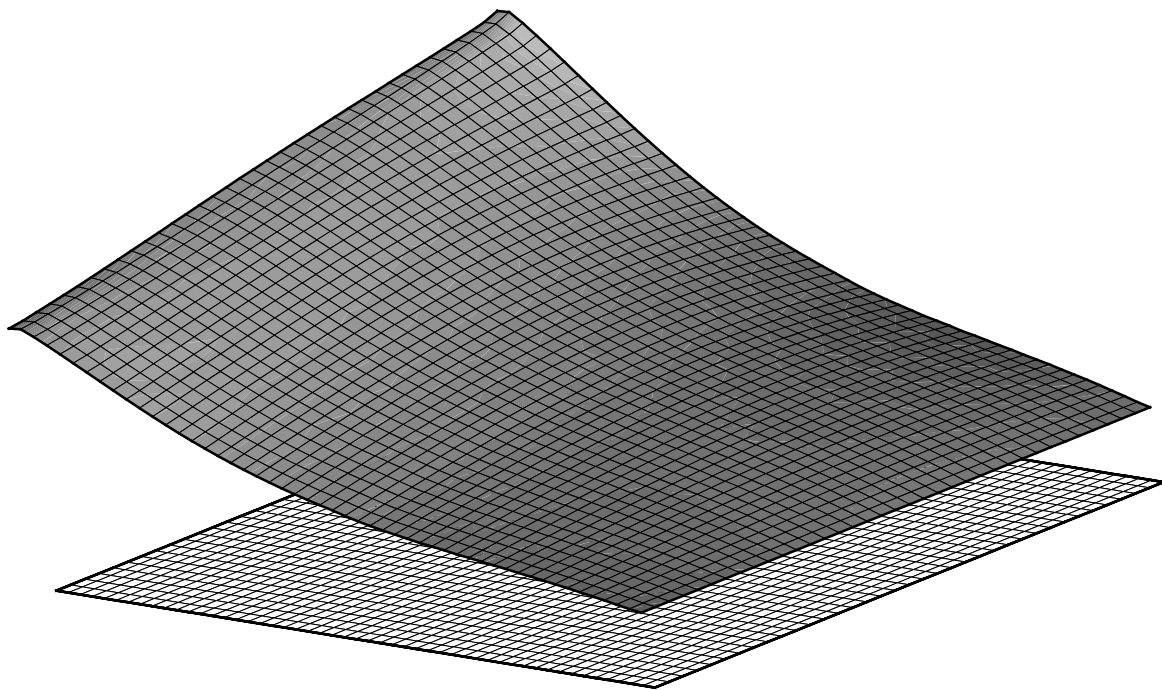


Figure 15: Fully blended cell sizing surface based on defining edges in parametric space.

Method Comparison

As stated in the introduction, the most common way of calculating the cell size to use for orthogonality specification is TFI. There are two different TFI's that can be utilized. The first is two dimensional in parametric space with $(\xi, \eta, \Delta s)$ being the three variables. Although this seems straight forward, the effects of the edge functions are not smeared and a smooth blending does not result (figure 16).

The second type of TFI that can be used is in three dimensional parametric space. For this type of interpolation, it was stated earlier that if TFI is used, due to its dependency on grid distributions, grid lines emanating from the surface may be skewed. Because of these grid lines being skewed, the cell sizes based on the standard arc-length calculation, equation 4, will be much larger than if the grid line is orthogonal to the surface (figure 17). In turn this produces a much larger cell size because the method is trying to stretch a grid line between two opposing faces with different distributions which arise from trying to model the flow field characteristics.

On the other hand, if the dependency on the grid distribution is removed and a smooth blending function is used between the defining cell sizes on the surface's edges, smaller initial grid spacings result (figures 16 and 17). In addition, smaller cell sizing gradients are produced across the surface domain. This occurs simply due to the fact that the orthogonality is assumed in the arc-length calculation. Therefore to obtain the best distribution of the cell sizes over the surface of the configuration, the LARCS method should be the method of choice. In doing so, the grid spacing near the surface will be smaller, lending to better and quicker resolution of the boundary layer in the evolution of the CFD solution.

As an example of the LARCS method, two different configurations are considered. The first configuration is the Space Shuttle (figure 18). By viewing the comparison of cell sizes in the contoured plots, the areas of highest cell sizes are the discontinuous regions. More specifically the wing root and fuselage juncture and the Orbital Maneuvering System (OMS) pod. This is also evident when viewing the resolved grid using both methods (figure 19). For the TFI grid, at the first grid point from the wing root and fuselage juncture, the LARCS method has placed 3 more grid points in the same relative distance. These extra grid points placed by the LARCS method and the elliptic solution, shows just how much more of the boundary layer can be resolved.

Also consider the Langley lifting body, the HL-20 (figure 20). Here the cell sizes, shown in figure 21, are similar between the two different methods. This can be attributed to the fact that the surface has no discontinuous regions. Yet, when considered more closely, at those regions of high curvature gradients, such as the wing root and fuselage juncture, the surface cell sizes do differ slightly. The sensitivity to the surface curvature is another condition that is neglected by the LARCS method. Furthermore, due to the smooth blending of the LARCS method, the gradients found on the fuselage forward of the wing for the three dimensional TFI, are not apparent for the LARCS method of distribution. Thus, the cell sizes based on the LARCS method are more likely to result in better resolution of the boundary layer, and not cause instability in the CFD solver, due to the smooth transitions between all cell sizing edge functions.

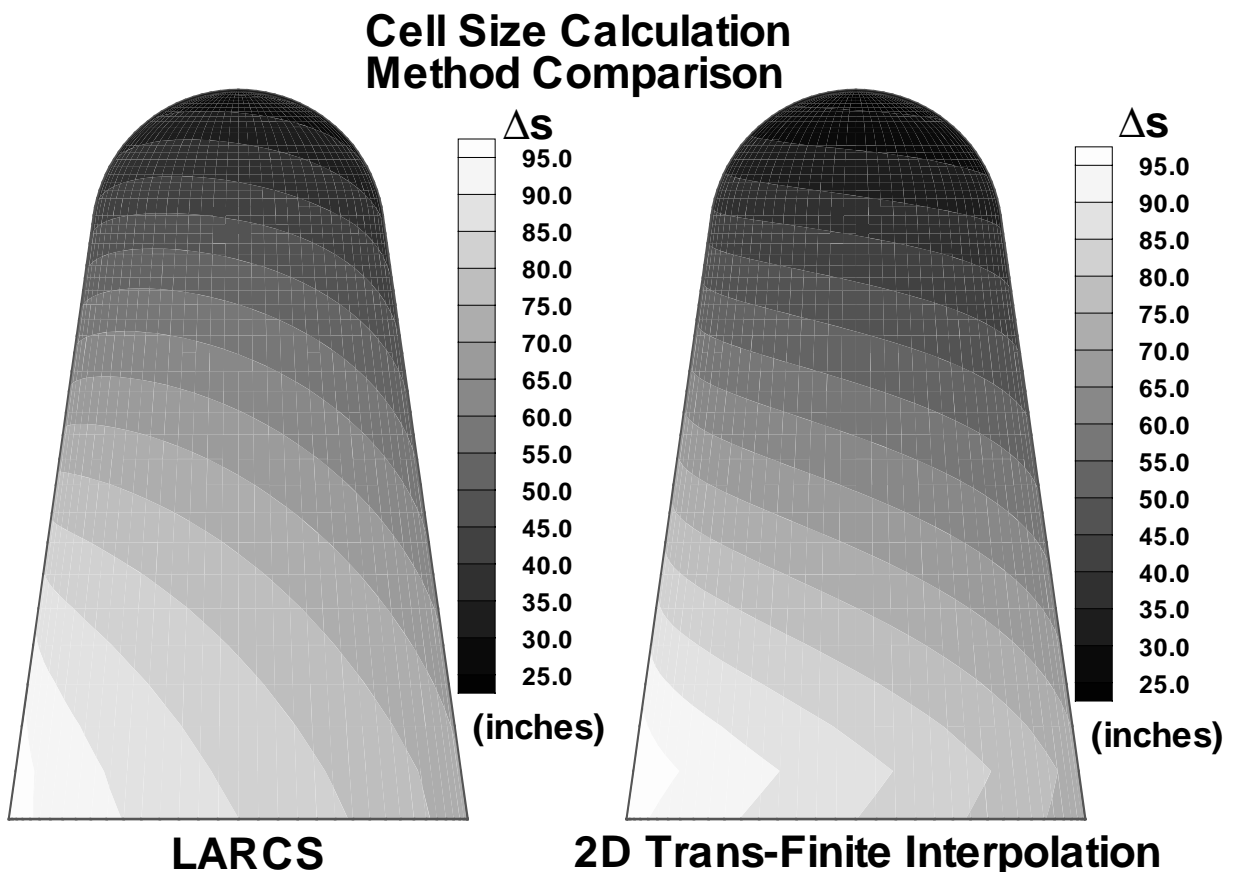


Figure 16: Cell sizing comparison of 2D TFI and the LARCS methods.

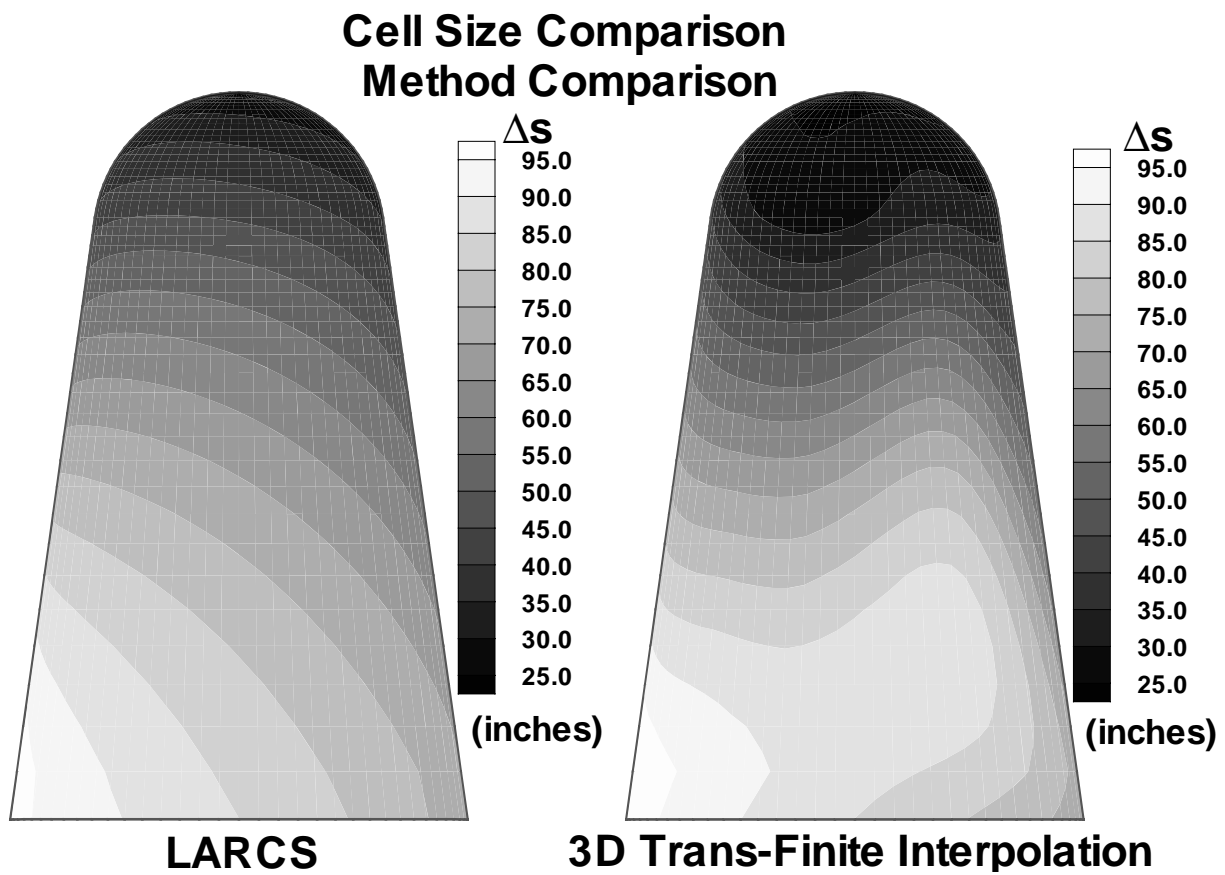


Figure 17: Cell sizing comparison of 3D TFI and the LARCS methods.

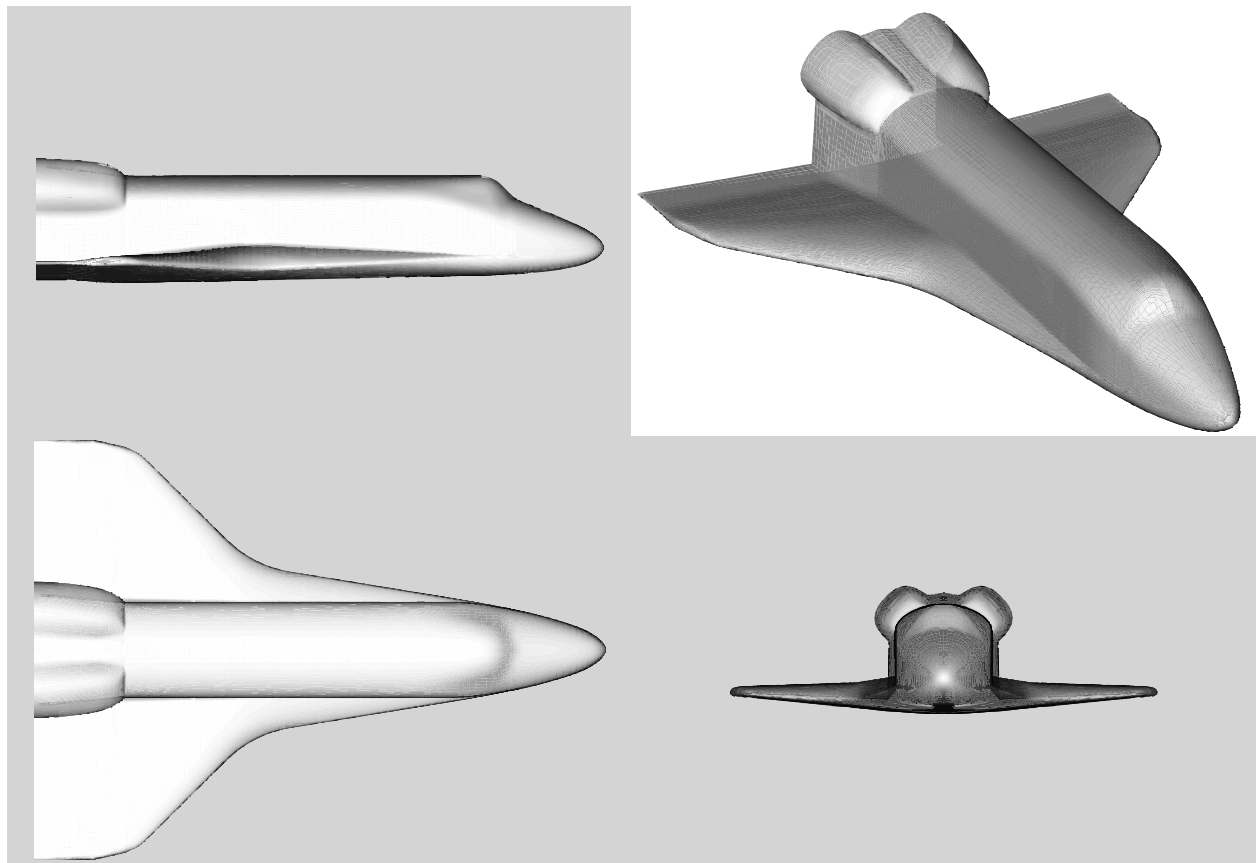


Figure 18: Space Shuttle surface geometry.

**Cell Size Calculation
Method Comparison
Shuttle Columbia**

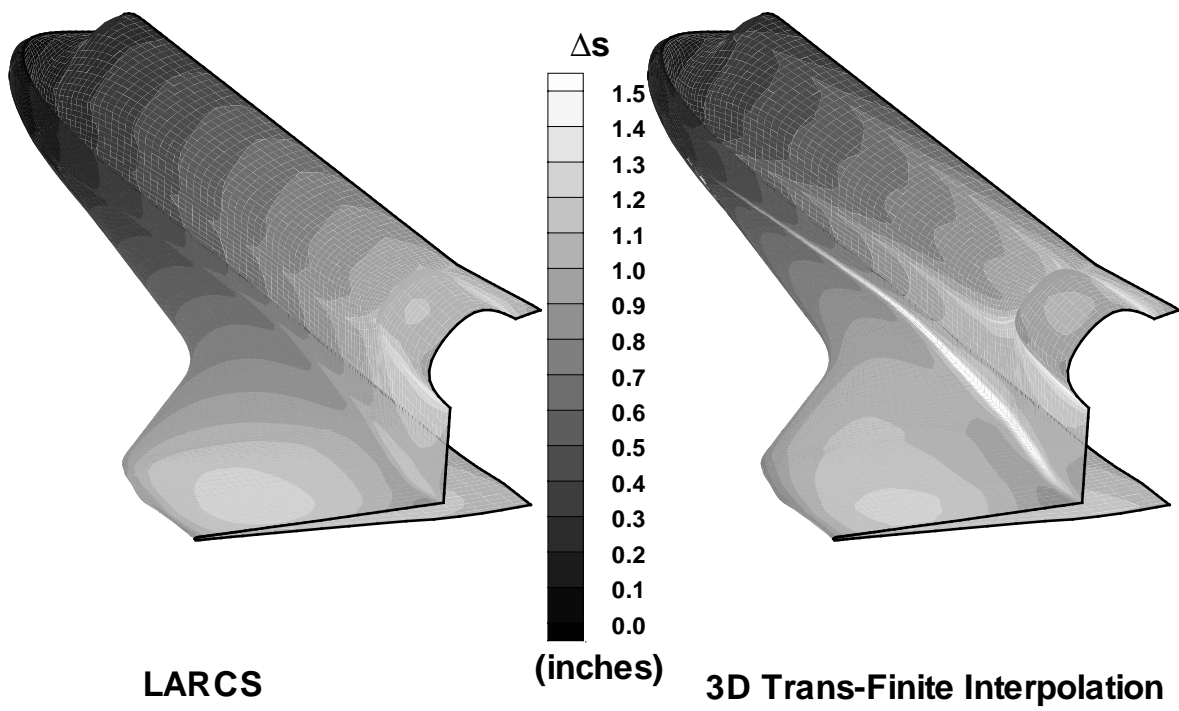


Figure 19: Cell sizing comparison of 3D TFI and LARCS for the Space Shuttle.

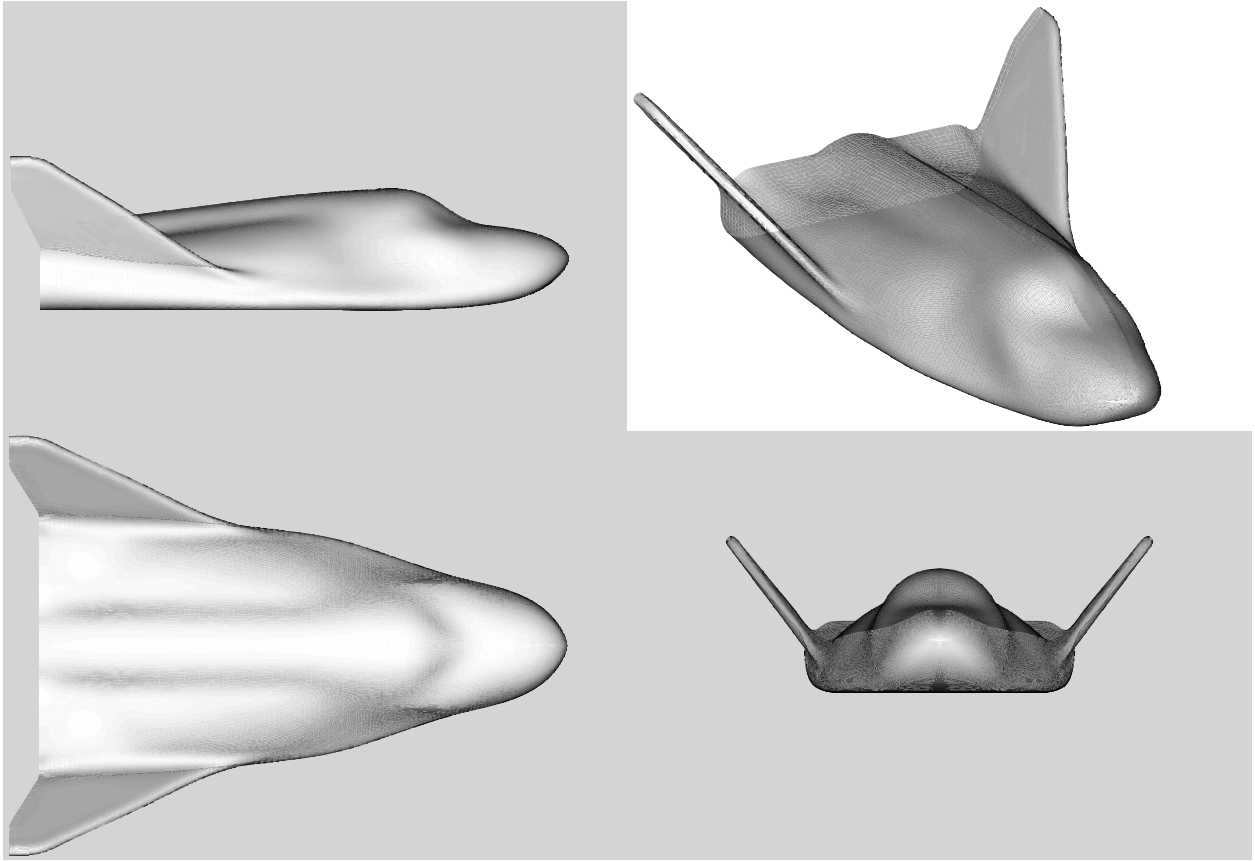


Figure 20: Langley Lifting Body (HL-20).

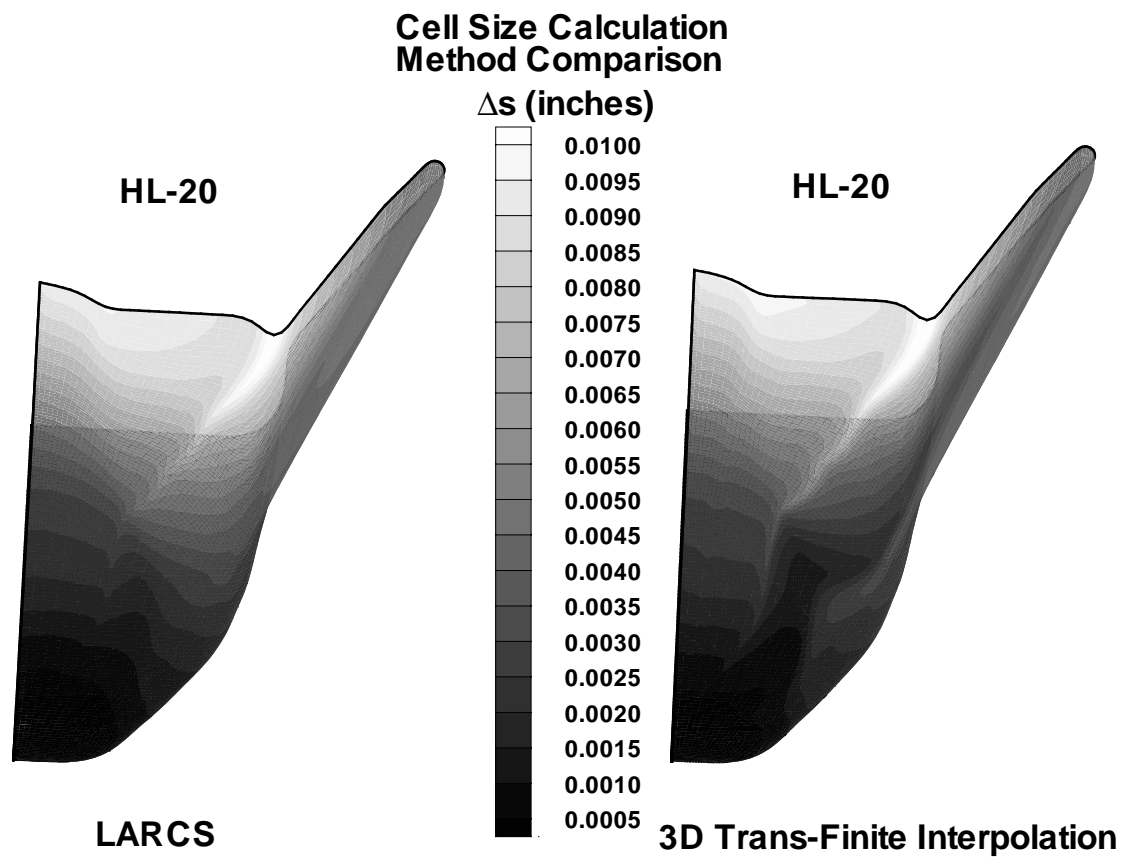


Figure 21: Cell sizing comparison of 3D TFI and LARCS for the HL-20.

Summary

An alternate method for calculating cell sizes for the explicit control of orthogonality in solving Poisson's three dimensional space equations has been presented. This method inherently builds a better link from the user's "best" guess for proper cell sizes to the specification of the cell sizes throughout the domain of the configuration's surface. The resulting cell sizes are much better than those generated by other methods, simply because LARCS eliminates any dependency on grid distributions, and is only concerned with the cell sizing distributions on the defining edges of the configuration's surface. In addition, the resulting cell sizes have smoother distributions on the surface of the configuration, which tends to reduce CFD solver inaccuracy, as opposed to large cell sizing gradients. Furthermore, by utilizing the LARCS method, the number of points near the surface will tend to be higher, and the flow solvers used will have a better chance of resolving the boundary layer. Finally, by having better resolution of the viscous forces before the first grid adaptation, the number of successive re-adaptations necessary to obtain a converged solution should be reduced. Thus, the user can obtain a quicker resolution of the boundary layer and the resulting converged solution, due to better initial grid spacing from the surface.

References

- [1] Reding, P. J. , Kudlick, D. A., "Aerodynamic Design Methods for Advanced Maneuvering Reentry Bodies," AIAA Missile Sciences Conference, November 13-15, 1990, Monterey, Ca.
- [2] Danberg, J. E., Sigal, A., Celmins, I., "Aerodynamic Characteristics of a Family of Cone-Cylinder-Flare Projectiles," AIAA paper 89-3371, August 1989.
- [3] Cruz, C. I., Englund, W. C., "Enhancements to the High Speed Convective Heating and Viscous Drag Prediction Techniques of the Aerodynamic Preliminary Analysis System (APAS)," AIAA paper 91-1435, June 24-26, 1991, Honolulu, Hawaii.
- [4] Steger, J. L., "Application of Mesh Generation to Complex 3-D Configurations" AGARD Advisory Report AR-268, March 1991.
- [5] Blottner, F. G., "Accurate Navier-Stokes Results for the Hypersonic Flow over a Spherical Nosedtip," AIAA paper 89-1671, June 1989.
- [6] Edwards, T. A., Flores, J., " Computational Fluid Dynamics Nose-to-Tail Capability: Hypersonic Unsteady Navier-Stokes Code Validation," AIAA paper 89-1672, June 1989.
- [7] Bhutta, B. A., Lewis, C. H., "Comparison of Hypersonic Experiments and PNS Predictions Parts I & II." *Journal of Spacecraft and Rockets*, Vol. 28, July-August 1991, pp. 376-393
- [8] Williams, R. M., "National Aerospace Plane: Technology for America's Future," *Aerospace America*, Vol. 24, No. 11, 1986, pp. 18-22

- [9] Olson, L. E., "Applied Aerodynamics," *Aerospace America*, Vol. 29, No. 12, pp. 60
- [10] Davies, C. and Venkatapathy, E., "A Simplified Self-Adaptive Grid Method, SAGE," NASA TM-102198
- [11] Gnoffo, P. A., "An Upwind-Biased, Point-Implicit Relaxation Algorithm for Viscous, Compressible Perfect-Gas Flows," NASA TP-2953
- [12] Anderson, J. D. Jr., "Modern Compressible Flow with Historical Perspective," McGraw-Hill, 1982, pp. 84-113, 248-259
- [13] Gupta, R. N., Lee, K., Thompson, R. A., Yos, J. M., "Calculations and Curve Fits of Thermodynamic and Transport Properties for Equilibrium Air to 30,000 K," NASA Reference Publication 1260, October 1991, pp. 14
- [14] Gupta, R. N., Lee, K., Thompson, R. A., Yos, J. M., "Calculations and Curve Fits of Thermodynamic and Transport Properties for Equilibrium Air to 30,000 K," NASA Reference Publication 1260, October 1991, pp. 59
- [15] Gnoffo, P. A., "An Upwind-Biased, Point-Implicit Relaxation Algorithm for Viscous, Compressible Perfect-Gas Flows," NASA TP-2953 pp. 15
- [16] Anderson, D. A., Tannehill, J. C., Pletcher, R. H., "Computational Fluid Mechanics and Heat Transfer," McGraw-Hill, 1984, pp. 343-346
- [17] Steinbrenner, J. P., Chawner, J. R., Fouts, C. L., "The GRIDGEN 3D Multiple Block Grid Generation System," Wright Research & Development Center, WRDC-TR-90-3022, October 1989.
- [18] Product Specification, PATRAN Plus, PDA Engineering, 1991
- [19] Sorenson, R. L., "The 3DGRAPE Book: Theory, User's Manual, Examples," NASA TM-102224, pp. 75-79